PCT

WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 **:** **G06F 13/14, 13/10, H04L 12/18, 12/58**	**A1**	**(11) International Publication Number:** **WO 00/10095** **(43) International Publication Date:** 24 February 2000 (24.02.00)

(21) International Application Number: PCT/US99/18346

(22) International Filing Date: 13 August 1999 (13.08.99)

(30) Priority Data:
09/134,737 14 August 1998 (14.08.98) US

(71) Applicant *(for all designated States except US)*: INTEL CORPORATION [US/US]; 2200 Mission College Boulevard, Santa Clara, CA 95052-8119 (US).

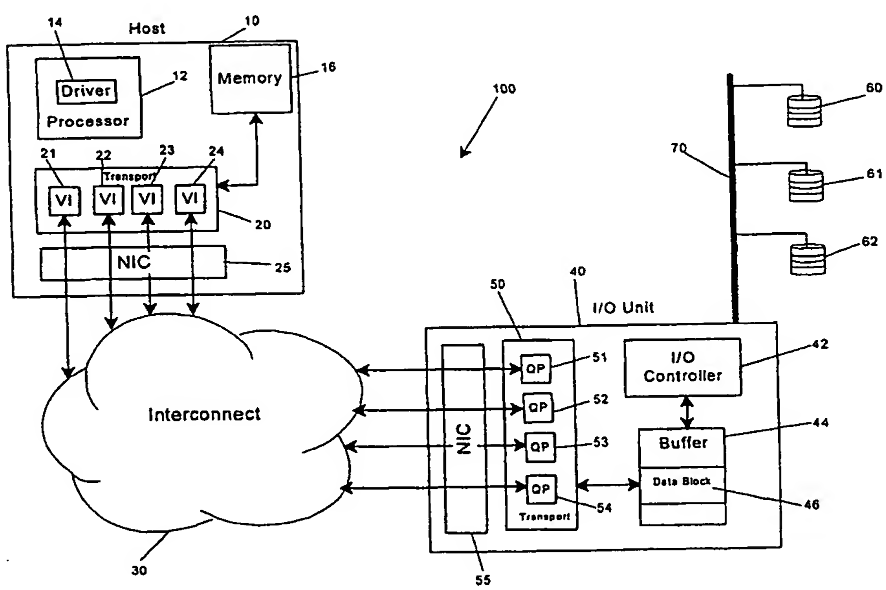
(72) Inventors; and
(75) Inventors/Applicants *(for US only)*: GRUN, Paul, A. [US/US]; 11852 S.W. Aspen Ridge Drive, Tigard, OR 97224 (US). FUTRAL, William [US/US]; 17715 N.W. Elk Run Drive, Portland, OR 97229 (US).

(74) Agents: SKWIERAWSKI, Paul, J. et al.; Antonelli, Terry, Stout & Kraus, LLP, Suite 1800, 1300 North Seventeenth Street, Arlington, VA 22209 (US).

(81) Designated States: AE, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CR, CU, CZ, DE, DK, DM, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, UA, UG, US, UZ, VN, YU, ZA, ZW, ARIPO patent (GH, GM, KE, LS, MW, SD, SL, SZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Published
With international search report.

(54) Title: STRIPING PACKETS OF DATA ACROSS MULTIPLE VIRTUAL CHANNELS



(57) Abstract

An I/O unit (40) for transporting a data block (46) having a plurality of data packets (147–150) across an interconnect (30) includes an I/O controller (42) and a memory (44) coupled to the I/O controller (42) for storing the data block (46). The I/O unit (40) further includes a DMA object (80) created by the controller (42) and referring to the data block (46), and a transport (50) that has a first and second VI queue pair (51, 52) being coupled to the interconnect (30). The I/O unit (40) further includes a first descriptor (90) created by the transport (50) and referring to a first data packet (147), and a second descriptor (91) created by the transport (50) and referring to a second data packet (148).

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania
AM	Armenia
AT	Austria
AU	Australia
AZ	Azerbaijan
BA	Bosnia and Herzegovina
BB	Barbados
BE	Belgium
BF	Burkina Faso
BG	Bulgaria
BJ	Benin
BR	Brazil
BY	Belarus
CA	Canada
CF	Central African Republic
CG	Congo
CH	Switzerland
CI	Côte d'Ivoire
CM	Cameroon
CN	China
CU	Cuba
CZ	Czech Republic
DE	Germany
DK	Denmark
EE	Estonia
ES	Spain
FI	Finland
FR	France
GA	Gabon
GB	United Kingdom
GE	Georgia
GH	Ghana
GN	Guinea
GR	Greece
HU	Hungary
IE	Ireland
IL	Israel
IS	Iceland
IT	Italy
JP	Japan
KE	Kenya
KG	Kyrgyzstan
KP	Democratic People's Republic of Korea
KR	Republic of Korea
KZ	Kazakstan
LC	Saint Lucia
LI	Liechtenstein
LK	Sri Lanka
LR	Liberia
LS	Lesotho
LT	Lithuania
LU	Luxembourg
LV	Latvia
MC	Monaco
MD	Republic of Moldova
MG	Madagascar
MK	The former Yugoslav Republic of Macedonia
ML	Mali
MN	Mongolia
MR	Mauritania
MW	Malawi
MX	Mexico
NE	Niger
NL	Netherlands
NO	Norway
NZ	New Zealand
PL	Poland
PT	Portugal
RO	Romania
RU	Russian Federation
SD	Sudan
SE	Sweden
SG	Singapore
SI	Slovenia
SK	Slovakia
SN	Senegal
SZ	Swaziland
TD	Chad
TG	Togo
TJ	Tajikistan
TM	Turkmenistan
TR	Turkey
TT	Trinidad and Tobago
UA	Ukraine
UG	Uganda
US	United States of America
UZ	Uzbekistan
VN	Viet Nam
YU	Yugoslavia
ZW	Zimbabwe

STRIPING PACKETS OF DATA ACROSS MULTIPLE VIRTUAL CHANNELS

FIELD OF THE INVENTION

The present invention is directed to data communication on a computer network. More particularly, the present invention is directed to striping packets of data across multiple virtual channels within a computer network.

BACKGROUND OF THE INVENTION

A computer frequently communicates with an input/output ("I/O") unit. The communication typically entails sending data across an interconnect at a high speed.

One of the most significant problems confronting this high-speed communication is the magnitude of processor and software overhead normally required. For example, a processor in the computer may need to dynamically

configure the bandwidth of the data transfer based on available resources in order to efficiently transfer the data. This process requires large software overhead which can prevent the processor in the computer, or an I/O controller in the I/O unit, from performing other tasks in a timely manner.

Based on the foregoing, there is a need for an method and apparatus to efficiently transfer data between a computer and an I/O unit.

SUMMARY OF THE INVENTION

One embodiment of the present invention is an I/O unit for transporting a data block having a plurality of data packets across an interconnect. The I/O unit includes an I/O controller and a memory coupled to the I/O controller for storing the data block. The I/O unit further includes a DMA object created by the controller and referring to the data block, and a transport that has a first and second VI queue pair, with each queue pair being coupled to the interconnect. The I/O unit further includes a first descriptor created by the transport and referring to a first data packet, and a second descriptor created by the transport and referring to a second data packet.

BRIEF DESCRIPTION OF THE DRAWINGS

Fig. 1 is block diagram of a networked computer system in accordance with one embodiment of the present invention.

Fig. 2 is a detailed block diagram of an I/O unit.

DETAILED DESCRIPTION

One embodiment of the present invention combines multiple physical channels into a single logical channel that can be used to transport data between an I/O unit and a host computer. An I/O unit includes one or more I/O controllers and an interface to an interconnect.

Fig. 1 is block diagram of a networked computer system in accordance with one embodiment of the present invention. The computer system 100 includes a host computer 10 and an I/O unit 40 coupled by an interconnect 30.

Interconnect 30 functions as an interface between networked devices. In one embodiment, interconnect 30 is a collection of switched connections that is created by combining multiple switches together. However, interconnect 30 can be any known infrastructure for connecting together networked devices. For example, interconnect 30 can comprise one or more direct connections (e.g., a wire, a local area network, or any other type of network), or one or more variable and dynamic connections (e.g., switches). Other networked devices (not shown in Fig. 1) that are included in computer system 100 could also be coupled to interconnect 30.

Host computer 10 includes a processor 12. Processor 12 executes a software application that includes a driver 14. Host 10 further includes a memory 16 and a transport 20. Host 10 further includes a network interface card ("NIC") 25 that couples host 10 to interconnect 30.

Host computer 10 communicates with devices coupled to interconnect 30 using a Virtual Interface ("VI") architecture. A VI architecture provides the illusion of a dedicated network interface to multiple applications and processes simultaneously, thus "virtualizing" the interface. Further, a VI architecture defines a standard interface between a VI consumer and one or more networks. In the present invention, driver 14 functions as a VI consumer.

In one embodiment, the VI architecture used to implemented the present invention is disclosed in the Virtual Interface Architecture Specification, Version 1.0, (the "VI Specification") announced December 19, 1997 by Compaq Corp., Intel Corp., and Microsoft Corp. The VI Specification is available at web site http://www.viarch.org/ on the Internet. The VI Specification defines mechanisms for low-latency, high-bandwidth message-passing between interconnected nodes and interconnected storage devices. Low latency and sustained high bandwidth are achieved by avoiding intermediate copies of data and bypassing the operating system when sending and receiving messages. Other architectures that perform a similar function as the VI architecture disclosed in the VI Specification can also be used to implement the present invention, and therefore the present invention is not limited to the VI architecture.

Transport 20 includes a plurality of VIs 21-24. Each VI 21-24 includes a queue pair ("QP"). In accordance with the VI Specification, a QP includes a send queue and a receive queue. In one embodiment, each VI 21-24 has a physical port

into interconnect 30 via NIC 25. However, in other embodiments, VIs 21-24 can share physical ports.

I/O unit 40 includes an I/O controller 42, a buffer memory 44 and a transport 50. An I/O controller is a device which provides I/O services to one or more host computers. The I/O services include storing and retrieving data, and transferring data to other devices. In other embodiments, I/O unit 40 includes two or more I/O controllers 42. Buffer memory 44 is a storage area that is coupled to I/O controller 40 and can be located anywhere. In one embodiment, buffer memory 44 is a separate storage device. In other embodiments, buffer memory 44 is part of disk drives 60-62, or is connected via a network to I/O controller 42.

Transport 50 includes QPs 51-54. A QP in I/O unit 40 and a corresponding VI in host 10 are endpoints of a virtual channel through interconnect 30. Although computer system 100 includes the same number of QPs 51-54 as VIs 21-24, in other embodiments the number of VIs in host 10 can differ from the number of QPs in I/O unit 40. However, each virtual channel through interconnect 30 has as endpoints one VI and one QP.

I/O unit 40 further includes a NIC 55 that couples I/O unit 40 to interconnect 30. In one embodiment, each QP 51-54 has a physical port into interconnect 30 through NIC 55. However, in other embodiments, QPs 51-54 can share physical ports.

I/O unit 40 further includes a plurality of disk drives 60-62 coupled to a bus 70. Disk drives 60-62 store data that can be accessed by host 10. In other embodiments, instead of disk drives 60-62, I/O unit 40 can include any other device that stores and/or retrieves data, or receives and forwards data to other devices. For example, I/O unit 40 can include a CD ROM drive, a tape drive, a network interface to a local area network, etc.

Within computer system 100, driver 14 is referred to as an "initiator" because it initiates requests for I/O services. In contrast, I/O controller 42 is referred to as a "target" because it responds to I/O requests from initiators within computer system 100. I/O controller 42 responds to I/O requests by, for example, storing data on drives 60-62 or retrieving data from drives 60-62.

An I/O service request can be generated by driver 14. An example of an I/O service request is a request to read a number of blocks of data from a logical block address in some storage device and return the data to a memory location in host 10. Driver 14 stores the I/O request in a location of memory 16. In accordance with the VI specification, driver 14 posts a descriptor that refers to the I/O request (i.e., specifies the location in memory 16 where the I/O request is stored) to a send queue in transport 20. Driver 14 then rings a doorbell in NIC 25. The doorbell tells NIC 25 to look in the send queue for the descriptor. NIC 25 then fetches the descriptor and performs the task. The task places an I/O request message on interconnect 30 to be

transmitted. The receiving device (e.g., I/O unit 30) of the I/O request also has a NIC that receives the I/O request message from interconnect 30.

The I/O request message contains information specifying the location in host memory 16 to which the data is to be moved, and specifies the location on the disk drives 60-62 from which the data is to be fetched. The location in host memory 16 is specified with a virtual address memory handle pair in accordance with the VI specification. I/O controller 42 uses the information contained in the I/O request message to build descriptors to accomplish the actual data movement from I/O unit 40 to host 10.

One embodiment of the present invention provides an efficient transfer of data from I/O unit 40 to host 10. In operation, driver 14 sends an I/O request to I/O unit 40 to retrieve a block of data (data block 46) from disk drives 60-62. The request is passed to I/O controller 42. In one embodiment, I/O controller 42 retrieves the requested data block 46 from drives 60-62 and stores data block 46 in buffer 44.

Fig. 2 is a detailed block diagram of I/O unit 40 that illustrates the steps executed by I/O unit 40 once data block 46 is stored in buffer 44. The goal of I/O unit 40 is to move data block 46 into memory 16 of host computer 10.

First, I/O controller 42 creates a direct memory access ("DMA") object 80 that refers to data block 46 in buffer 44. DMA object 80 specifies the starting address of data block 46, the length of data block 46, and the destination of data block 46. The destination tells transport 50 where to send data block 46. The destination

includes the endpoint across interconnect 30 to which data block 46 is to be sent, and the memory address at that endpoint where data block 46 is to be stored.

I/O controller 42 passes DMA object 80 to transport 50. In one embodiment, transport 50, before receiving DMA object 80, has already created the correct number of QPs 51-54 based on the number of physical channels between I/O unit 40 and host 10. In another embodiment, transport 50 dynamically creates the necessary number of QPs upon receiving DMA object 80. In this embodiment, a network services unit is connected to interconnect 30 in Fig. 1. The network services unit is responsible for managing interconnect 30 and creating virtual connections in interconnect 30. Transport 50 specifies to the network services unit the required number of virtual channels based on the number of physical links, and then creates the corresponding QPs.

Transport 50 creates one or more descriptors 90-93 for each QP in transport 50. In one embodiment, each QP in transport 50 corresponds to a physical port into interconnect 30. In other embodiments, there can be less physical ports than QPs in transport 50. Each descriptor 90-93 describes a partitioned portion, or "packet" of data block 46. For example, descriptor 90 represents data packet 147, descriptor 91 represents data packet 148, etc. Packets 147-150 can be of varying size depending on the algorithm used to partition data block 46. For example, in one embodiment, each packet 147-150 is equal size. In another embodiment, an algorithm is used to

determine the size of each packet 147-150 based on the operating characteristics of the available physical connections between host 10 and I/O unit 40 in interconnect 30.

Each descriptor 90-93 is then posted into its respective QP 51-54. QPs 51-54 move the packet represented by its descriptor 90-93 to their physical port for transport across interconnect 30. Therefore, the packets are "striped" across multiple physical connections. The order that each descriptor 90-93 is moved across interconnect 30 is arbitrary.

In the embodiment shown in Fig. 2, all four QPs 51-54 run in parallel because they each have a separate physical connection. However, the multiple physical connections are abstracted from I/O controller 42. Therefore, it appears to I/O controller 42 and to driver 14 that there is one virtual channel running with four times the bandwidth of a single physical channel.

I/O controller 42 is only aware of creating a single DMA object 80 for the entire data block 46. Transport 50 implements the functionality of striping data block 46 over multiple physical channels, thus alleviating the overhead from I/O controller 42.

As described, the VI in transport 50 is used to give driver 14 or any other initiator in host 10 the illusion of an arbitrarily large bandwidth. Further, transport 50 combines multiple physical channels into one large logical channel without imposing any increased overhead on I/O controller 42 or processor 12.

Several embodiments of the present invention are specifically illustrated and/or described herein. However, it will be appreciated that modifications and variations of the present invention are covered by the above teachings and within the purview of the appended claims without departing from the spirit and intended scope of the invention.

For example, one additional embodiment of the present invention can involve the transfer of data between a host computer and another computer, instead of between a host computer and I/O unit 40. In this inter-processor communication ("IPC") environment, the other computer includes a processor that performs the functions of I/O controller 42. All other elements of the other computer are similar to I/O unit 40, and the data is transferred in an identical manner as described herein.

WHAT IS CLAIMED IS:

1. An input/output (I/O) unit for transporting a data block across an interconnect, the data block comprising a plurality of data packets, said I/O unit comprising:

an I/O controller;

a memory coupled to said I/O controller for storing the data block;

a direct memory access (DMA) object created by said controller and referring to the data block;

a transport having a first and second virtual interface (VI) queue pair, each queue pair coupled to said interconnect;

a first descriptor created by said transport and referring to a first data packet; and

a second descriptor created by said transport and referring to a second data packet.

2. The I/O unit of claim 1, wherein each queue pair is coupled to said interconnect through a physical port.

3. The I/O unit of claim 1, wherein said first descriptor is posted on said first queue pair and said second descriptor is posted on said second queue pair.

4. The I/O unit of claim 1, wherein said DMA object comprises:

a starting address of the data block;

a length of the data block; and

a destination of the data block.

5. The I/O unit of claim 1, further comprising a network interface controller coupled to said transport and the interconnect.

6. The I/O unit of claim 1, wherein said first data packet and said second data packet form the data block.

7. The I/O unit of claim 1, further comprising

a third descriptor created by said transport and referring to a third data packet;

wherein said first data packet, said second data packet, and said third data packet form the data block.

8. The I/O unit of claim 1, wherein said first and second queue pair each comprise a send queue and a receive queue.

9. The I/O unit of claim 1, further comprising:

a disk drive coupled to said I/O controller.

10. The I/O unit of claim 1, wherein a first virtual connection comprises said first VI queue pair and a virtual interface coupled to said interconnect.

11. A method of transporting a data block across an interconnect to a host computer, said method comprising:

(a) storing the data block in memory;

(b) creating a direct memory access (DMA) object that refers to the data block;

(c) partitioning the data block into a plurality of data packets;

(d) creating a plurality of descriptors, each of said descriptors referring to one of said data packets;

(e) posting said plurality of descriptors into virtual interface queue pairs, said queue pairs coupled to virtual channels that are coupled to the host computer; and

(f) moving said plurality of data packets across said interconnect on the virtual channels.

12. The method of claim 11, wherein the host computer comprises virtual interfaces and the virtual channels are coupled to the virtual interfaces.

13. The method of claim 11, wherein each of said data packets is equal size.

14. The method of claim 11, wherein said data packets differ in size.

15. The method of claim 11, wherein said DMA object comprises:

a starting address of the data block;

a length of the data block; and

a destination of the data block.

16. A computer for transporting a data block across an interconnect, the data block comprising a plurality of data packets, said computer comprising:

a processor;

a memory coupled to said processor for storing the data block;

a direct memory access (DMA) object created by said controller and referring to the data block;

a transport having a first and second virtual interface (VI) queue pair, each queue pair coupled to said interconnect;

a first descriptor created by said transport and referring to a first data packet; and

a second descriptor created by said transport and referring to a second data packet.

17. A networked computer system comprising:

an interconnect;

a host computer coupled to said interconnect, said host computer comprising a first transport having a first and second virtual interface;

an I/O unit coupled to said interconnect, said I/O unit comprising

an I/O controller;

a first memory coupled to said I/O controller for storing a data block;

a direct memory access (DMA) object created by said controller and referring to the data block;

a second transport having a first queue pair coupled to said first virtual interface to form a first virtual channel and a second queue pair coupled to said second virtual interface to form a second virtual channel;

a first descriptor created by said second transport and referring to a first data packet; and

a second descriptor created by said second transport and referring to a second data packet.

18. The networked computer system of claim 17, wherein said first and second queue pairs are coupled to said interconnect through a physical port.

19. The networked computer system of claim 17, wherein said first descriptor is posted on said first queue pair and said second descriptor is posted on said second queue pair.

20. The networked computer system of claim 17, wherein said DMA object comprises:

a starting address of the data block;

a length of the data block; and

a destination of the data block.

22. The networked computer system of claim 17, said I/O unit further comprising a network interface controller coupled to said second transport and said interconnect.

23. The networked computer system of claim 17, wherein said first data packet and said second data packet form the data block.

24. The networked computer system of claim 17, wherein said first and second queue pair each comprise a send queue and a receive queue.

25. The networked computer system of claim 17, said I/O unit further comprising:

a disk drive coupled to said I/O controller.

26. The networked computer system of claim 17, said host computer further comprising:

a processor executing a driver; and

a second memory coupled to said processor for storing said first data packet and said second data packet when received from said I/O unit.

27. The networked computer system of claim 19, wherein said driver is an initiator and said I/O controller is a target.

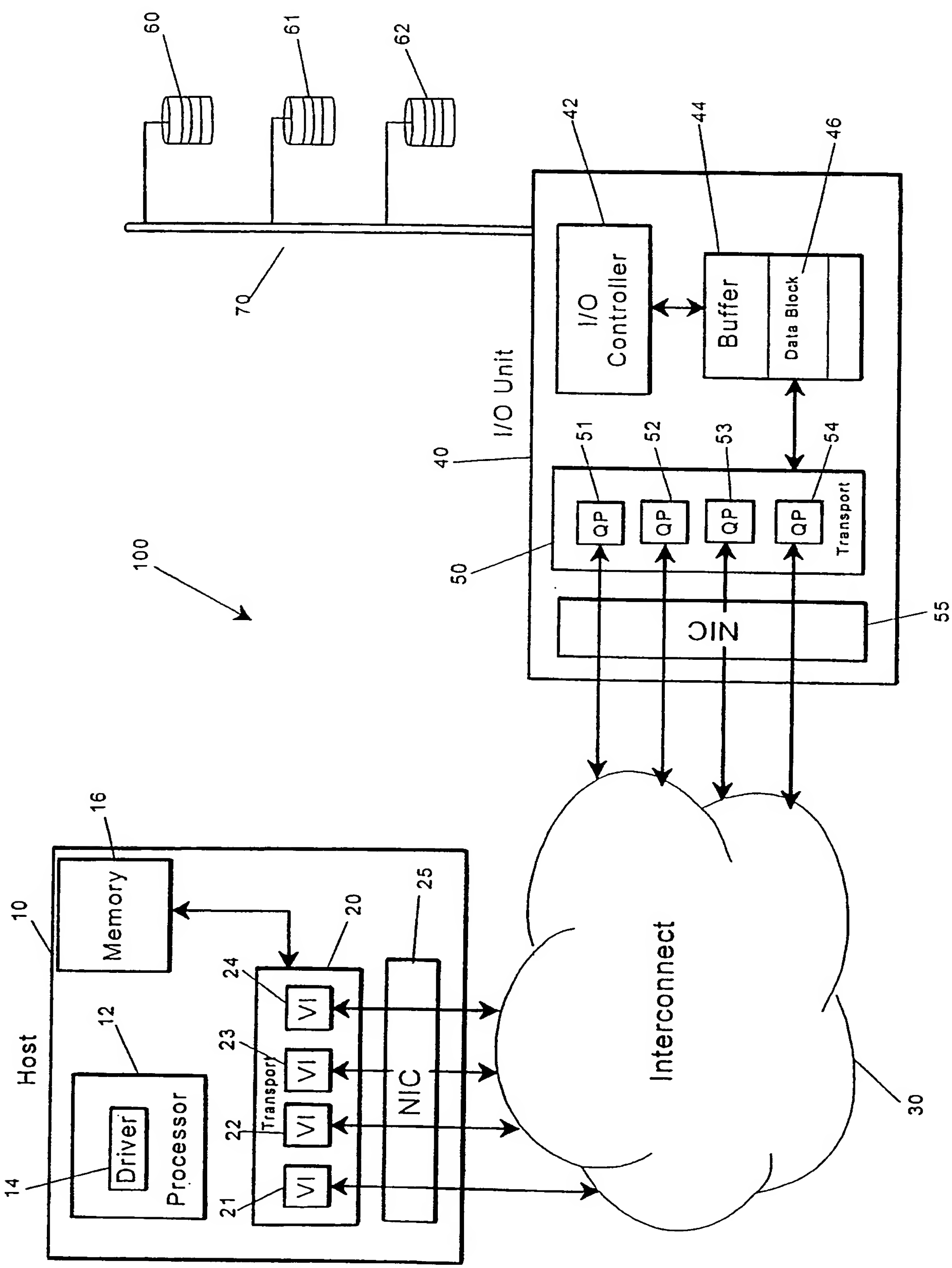

Host
Processor
Driver
Memory
Transport
VI
VI
VI
VI
NIC
Interconnect
I/O Unit
I/O Controller
Buffer
Data Block
QP
QP
QP
QP
Transport
NIC
10
12
14
16
20
21
22
23
24
25
30
40
42
44
46
50
51
52
53
54
55
60
61
62
70
100

Fig. 1

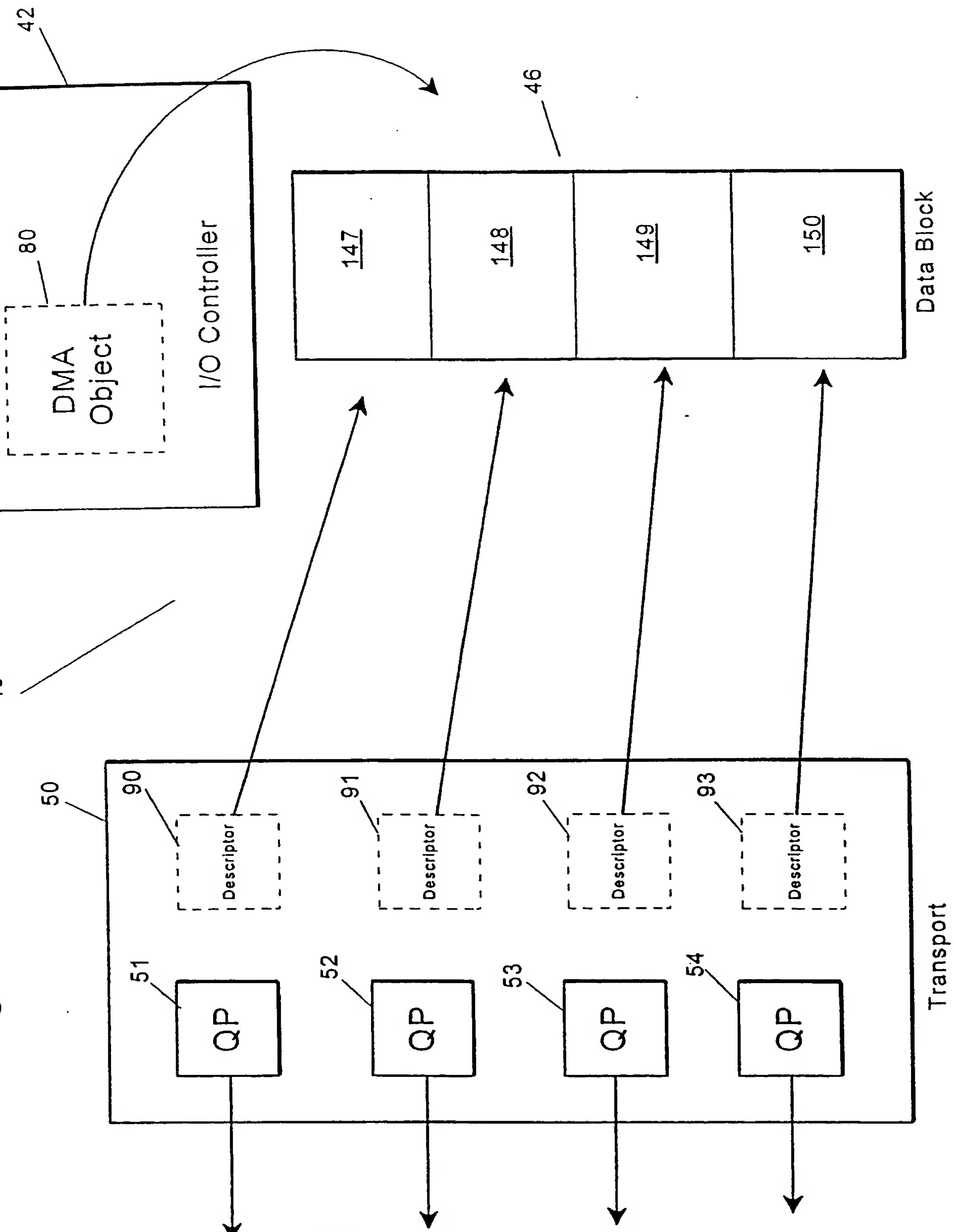

42
80
DMA
Object
I/O Controller
46
147
148
149
150
Data Block
40
50
90
91
92
93
Descriptor
Descriptor
Descriptor
Descriptor
51
52
53
54
QP
QP
QP
QP
Transport

Fig. 2

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US99/18346

A. CLASSIFICATION OF SUBJECT MATTER

IPC(6) :G06F 13/14, 13/10; H04L 12/18, 12/58
US CL :Please See Extra Sheet.
According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

U.S. : 370/235, 397, 399, 412, 464, 473; 709/250; 710/29, 33; 711/112, 114

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

APS, WEST
search terms: virtual interface, virtual connection, physical connection, receive queue, send queue, packet transfer

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 5,787,086 A (MCCLURE et al) 28 July 1998, col. 4, lines 36-59, col. 5, lines 13-15, col. 6, lines 26-67, col. 8, lines 15-20	1-27
Y	US 5,745,684 A (OSKOUY et al) 28 April 1998, col. 6, lines 37-48, col. 8, lines 24-27	1-27
Y	US 5,577,033 A (CHANG et al) 19 November 1996, col. 4, lines 14-26, col. 8, lines 12-20	6, 7, 23

☐ Further documents are listed in the continuation of Box C. ☐ See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"B" earlier document published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
06 OCTOBER 1999	26 OCT 1999

Name and mailing address of the ISA/US	Authorized officer
Commissioner of Patents and Trademarks Box PCT Washington, D.C. 20231	FOR THOMAS C. LEE
Facsimile No. (703) 305-3230	Telephone No. (703) 305-9717

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US99/18346

A. CLASSIFICATION OF SUBJECT MATTER:
US CL :

370/235, 397, 399, 412, 464, 473; 709/250; 710/29, 33; 711/112, 114

Form PCT/ISA/210 (extra sheet)(July 1992) ★